Design Proposal for IBL's DSP ALU

Robert W. Costanzo Jr.
rwc3bf@virginia.edu

Austin Moran
apm4yw@virginia.edu

Hector L. Soto Jr.
hls6dc@virginia.edu

ECE 3663 – Spring 2013
University of Virginia

1. INTRODUCTION

The purpose of the design project was to ultimately design an 8-bit DSP ALU that would meet and exceed the expectations of the Itty Bitty Logic Co. The client IBL requested that the arithmetic logic unit support eight unique functions, one of which was up to the designers to choose and implement. Throughout the design process, our company simulated multiple designs in order to verify proper operation and was subjected to several design reviews, all yielding positive results and feedback. Numerous metrics specified by IBL were taken into consideration during design and implementation. These metrics included delay, maximum reliable clock frequency, area, and energy dissipation. Our company believes that our design for the desired logic block surpasses IBL's prescribed specifications and guarantee a reliable and effective product.

2. DESIGN DESCRIPTION

As far as the upper level logic is concerned, the DSP path was kept almost identical to the IBL-prescribed layout. In the simplest sense, our ALU ended up consisting of blocks of digital logic for each desired function whose outputs were then tied to the inputs of an eight-to-one multiplexer. These functions included logic blocks for AND, OR, XOR, ADDER, 2's COMP, PASS-A, and NO-OP (PASS-B). Originally, there was a SHIFTER but our three-person group was not responsible for it. Inputs to the ALU consisted of an external input A, and input B created by feeding the output of the ALU back as a second input. To support large adder inputs, the ALU also contained an carry-out signal. The inputs, outputs, and select lines for the MUX were all loaded into registers to ensure that the signals would transition on the same rising-edge clock cycle and prevent unnecessary yet detrimental glitches in signal propagation.

3. INNOVATION

Our company believes that the optimizations and innovative ideas utilized in the design of the DSP ALU distinguish our proposal from IBL's numerous other options. Static CMOS logic was initially used, and then transmission gate logic was incorporated in each logic block based on a case-by-case analysis. Our primary focus in design was decreasing the number of transistors used without sacrificing reliable operation. It was our belief that decreasing this metric directly improved other metrics as well, such as delay, size, and energy dissipation.

The Multiplexer

During the early stages of implementation, the company agreed to focus primarily on functionality. As such, the eight-to-one multiplexer was designed using static CMOS logic. Essentially, we used standard digital logic design (implementation using separately created gates - ie. boolean logic) and combined each input with a combination of select lines into a 4-input NAND. All these NAND's ran into a final 8-input NAND. Logically, this worked as desired, but the large NAND gates created unnecessary delay and consisted of a large amount of transistors. However, we then optimized the design and decided to utilize a transmission gate logic approach. This reduced the number of transistors from 86 to 54 and dramatically lowered delay and area. The critical path no longer involved two large NAND gates like most, but only a series of three transmission gates.

The AND Gate

The AND gate did not respond well to being implemented in transmission gates. The proposed Tgate design would have only used a total of five transistors. This would have minimized the number of transistors and kept the power usage low. However, the t-gate design did not have good delay characteristics. A simulated graph of the output showed nearly double the delay from input to output of a CMOS design and displayed several glitches (temporary errors) when the output changed states. For these reasons the t-gate design was abandoned and CMOS was used in its place. We chose to maintain the integrity of the signals over saving one transistor in the gate.

The OR Gate

The OR gate was not suitable for implementation using t-gates or pass transistor logic. PTL didn't work on account of the threshold voltage drop being 0.7V and the supply voltage being 1.1V (1.1V - 0.7V = 0.4V which is below 1.1V/2 = 0.55V). It was possible however to use a series of two inverters to restore the signal, but would add more transistors. This would have defeated the purpose of attempting the PTL or t-gate logic and would have actually increased transistors used. Therefore, we decided to implement the design with the fewest transistors without sacrificing signal quality.

The XOR Gate

We decided to use the XOR gate as our arbitrary function based on its usefulness as an operator. The original XOR gate used standard DLD logic and was comprised of four NAND gates. However, we turned it into a more efficient static CMOS logic circuit. With this change, we were able to reduce the number of transistors used in the XOR gate from 16 transistors to

12 transistors without sacrificing reliability. The delays were fairly comparable, and the change mostly impacted area. A reduction in the XOR gate also resulted in a reduction in the Adder size (discussed next).

The Adder

At the 1-bit level, our company decided to use the full adder which, while it was slightly larger, gave the ALU more utility. With the original digital logic design the adder consisted of 56 transistors. However, we looked into transmission gate designs for full adders and after testing, decided that an implementation using 16 transistors was still capable of providing reliable signal at a fast speed. At the 8-bit level, the company decided to implement a ripple-carry adder instead of other typologies. We found that while the ripple-carry adder yields a slow critical path delay, we had already decided to prioritize size. Other adder implementations, such as the carry-lookahead adder, required additional hardware such as multiplexers and control lines that significantly increased the size of the proposed ALU design. Again, as we focused on saving size and cost, we decided to accept the extended critical path.

Trade-Offs

Ultimately, the our company would focus on size and number of transistors as our optimization metric. We strived to reduce the area of the ALU DSP to reduce costs while refusing to sacrifice signal quality and strength. However, this did not come without giving something up. Deciding not to implement additional hardware to construct faster hardware for the adder, the adder would become our maximum critical path. We believed that keeping our number of transistors low would lower costs for IBL and was our deciding factor.

4. RESULTS

We used area, delay, and energy as our main metrics for analysis.

Area

The total number of transistors in our original 8-bit ALU DSP totaled to 2496. However, after optimization, we were able to successfully reduce the number by 28.2 percent, down to 1792. The total size of the device widths are shown below:

Table 1: Count of transistors used in the proposed ALU DSP

	Number Used	Width/Device	Total Area
NMOS	896	90nm	80.64um
PMOS	896	180nm	161.28um

This means our total width of our device is 241.92um. Our company guarantees IBL that our design is about as small as possible while not sacrificing signal quality.

Power

We found through tests that our ALU DSP consumes an average of $3.781*10^{-17}$ J/cycle. This was based off of a suggested test scenario that involved changing signals as described at a frequency of 10MHz.

Delay

We found, as expected, that the worst case delay occurred in the carry ripple propagation path in the adder. We found this worst case delay to be approximately 462 ps. The delays of the other components are much smaller, and this maximum delay is measured from the first bit of the adder, through to the eighth bit via carry ripple propagation, and then through the MUX.

Table 2: Worst Case Delays from Individual Components

Device	Delay
AND	83 ps
OR	59.7 ps
PASS A/B	55.4 ps
XOR	92 ps
ADDER	220 ps
2's COMP	296.4 ps
MUX	76.9 ps

Maximum Clock Frequency

Due to the maximum critical delay (including registers setup and hold times), we have found that the maximum delay from input to output is 9.52 ns. Therefore, our company is confident that our processor is capable of running at 100MHz and as such, ran our final tests at that speed.

Final Metric Score

The metric for the design is Energy * Delay * Area = (241.92*10^-12 m^2)(9.52*10^-9 s)(3.781*10^-17 J) = 87.0794*10^-36 m^2 J s

5. CONCLUSIONS

Our design is efficient in several ways. The design uses fewer transistors than a static CMOS implementation due to optimizations involving transmission gates. Circuit optimizations decreased the total area of the design by 28.2%. The maximum delay of the design is 9.52 nanoseconds. This allows for a theoretical maximum clock cycle of ~105 MHz. 105MHz exceeds the requirement of the IBL for 100MHz or would allow for a nice (5%) safety margin on the running conditions of the DSP. Energy usage of the design is minimal, kept low by the usage of static CMOS circuits throughout. CMOS keeps energy used low because it only uses (effectively) energy when the output changes states (compared to Dynamic CMOS design). Some design choices and potential optimizations were discarded due to their potential for error. Special consideration was given to the output graphs of the transient simulations. Any graphs that showed glitches or other temporary errors in the output meant a discarded design. This was to ensure that the design is resilient against errors and thus potential loss of data or miscalculations. Any designs that didn't reach either VDD/VSS quickly were discareded as well for the same reasons. All of the

optimizations and design considerations have produced a design that is energy efficient, space and resource efficient (low area), robust against potential error, and meets the desired clock frequency of 100 MHz with a built in safety margin. Simply put, by choosing to optimize two of the metric components (Energy and Area) instead of all three, we have created the superior circuit (with regards to energy and size efficiency).

6. ACKNOWLEDGMENTS

Our thanks to ACM SIGCHI for allowing us to modify templates they had developed.